

**APPLICATION
FOR
UNITED STATES LETTERS PATENT**

Applicant:

David F. Bantz, et al.

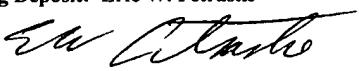
For:

“Internet Site Authentication Service”

Docket: YOR920030474US1

**INTERNATIONAL BUSINESS
MACHINES CORPORATION
ARMONK, NEW YORK 10504**

I HEREBY CERTIFY THAT THIS CORRESPONDENCE IS BEING DEPOSITED WITH THE
UNITED STATES POSTAL SERVICE AS EXPRESS MAIL IN AN ENVELOPE ADDRESSED
TO: U.S. PATENT AND TRADEMARK OFFICE, WASHINGTON, D.C. 20231 THE
APPLICANT AND/OR ATTORNEY REQUESTS THE DATE OF DEPOSIT AS THE FILING
DATE.

Express Mail No: EU700226785US
Date of Deposit: 11/16/03
Name of Person Making Deposit: Eric W. Petraske
Signature: 

Internet Site Authentication Service

1 **TECHNICAL FIELD**

2 The field of the invention is that of checking the identity of a web site on the Internet, in
3 particular verifying that the web site address has not been taken over by an imposter.

4 **BACKGROUND OF THE INVENTION**

5 There are millions of sites now on the Internet. Recently, the creators of some of these
6 sites are perpetrating a certain type of fraud. In this fraud, a site masquerades as another
7 site, typically a site engaged in e-commerce. The masquerader site has the same
8 appearance as the authentic site, but its programming is different. The masquerader may
9 simply pull business away from the authentic site, or it may capture personal information
10 about a user for nefarious purposes. The information may be just the user's Internet
11 address, or it may be obtained from the user via dialogs, and may include the user's credit
12 card number, social security number or any such. The masquerade is facilitated through
13 hijacking of some subset of the Internet's domain name service (DNS) that maps
14 Universal Resource Locators (URLs) to Internet addresses. When DNS is hijacked it
15 returns the address of the masquerader rather than the address of the legitimate site.

16 Increased assurance that a web site that the user is browsing is legitimate has value to
17 both the user and to the operator of the site. If a user is reasonably confident that a web

1 site is legitimate he or she will be more likely to supply that web site with personal or
2 business information of value and to conduct business with that site.

3 US 5,717,756 to Coleman addresses the problem of the authentication of one machine to
4 another. That approach uses the generation of a unique machine-specific key. The
5 problem of current concern is not machine-specific: any web server that serves legitimate
6 content is of equal value to the end user, and changes in machine configuration do not
7 affect this value.

8 In US 5,953,424 Vogelesang et. al. a cryptographic protocol is described capable of
9 "detection of prior occurrences of unauthorized parties successfully masquerading as an
10 authorized party." This protocol requires modification of both the site and the user's
11 browser to implement the protocol, while the subject invention requires no modification
12 of the site.

1 Solutions to this fraud are known, but have drawbacks. In one solution, the end user
2 comes into possession of a certificate from the site and validates that certificate, either
3 locally or with a third party. Certificate validation locally adds to the complexity and cost
4 of accessing the Internet. It does not apply if the user cannot or has not obtained the
5 certificate, or if the user's computer has lost the certificate, as it might during a recovery
6 process.

7 Third-party certificate validation adds a delay to the access to the desired site and may
8 add a cost if the third party charges for validation. It is even possible for a site to spoof
9 validation by registering with the third party, but subsequently to change its appearance
10 and function and hijack DNS so as to masquerade as another legitimate site.

11 It is desirable for any solution to this problem to be implemented in a transparent way that
12 is, without additional behavior that would be apparent to the end user. This is due, in part,
13 to the fact that masquerading is infrequent, although highly disruptive when it exists.

14 A solution must not complicate the implementation of a legitimate site nor increase its
15 resource requirements significantly.

16 It would be advantageous to users and site operators to be able to automatically
17 differentiate between legitimate and masquerader sites, alerting the end user
18 appropriately. This, in turn, increases the user's confidence that his or her transaction will
19 be with a legitimate site and in so doing, increases the business potential of the Internet.

20 SUMMARY OF THE INVENTION

1 The invention relates to methods for the automatic detection of masquerading web sites
2 and the alerting of an Internet user to that situation.

3 A feature of the invention is that the solution to this problem is implemented in a
4 transparent way, without additional behavior that would be apparent to the end user.

5 A feature of the invention is software that runs on a personal computing device and a
6 service that is provided via the Internet. The software on the personal computing device is
7 in the form of a proxy, or transparent component in the Internet Protocol implementation.
8 The proxy receives all outbound messages, analyzes them and forwards or modifies them
9 without the user's intervention.

10 Another feature of the invention is an authentication server that performs a test on the
11 target websites' IP address and behavior.

12 BRIEF DESCRIPTION OF THE DRAWINGS

13 Figure 1 shows an overall diagram of a system employing the method.
14 Figure 2 shows steps in the proxy software.
15 Figure 3 shows steps in the validation server.

1 **DETAILED DESCRIPTION**

2 According to the invention, the authenticity of Internet sites is verified by a combination
3 of software that runs on a personal computing device and a service that is provided via
4 the Internet. The software on the personal computing device is in the form of a proxy, or
5 transparent component in the Internet Protocol implementation. The proxy receives all
6 outbound messages, analyzes them and forwards or modifies them without the user's
7 intervention.

8 The proxy intercepts web site access requests and interacts with a web-based service to
9 validate the web site that these requests are directed to. The service makes an assessment
10 if the requests are to a legitimate web site. If they are, then they are either forwarded or
11 the proxy is notified with an approval message that the requests can be sent directly. If the
12 assessment is that the site is not legitimate, then the proxy is notified that the web site that
13 would have been accessed is probably a masquerader.

1 In a preferred embodiment, the authentication service employs a combination of analyses
2 of Internet addresses and site behavior. The service maintains knowledge equivalent to
3 that of the Internet domain name resolution service for specific web sites known to be
4 under masquerading attack.

5 If a domain name resolves to an Internet address different from that known to the service
6 as being the legitimate site, this is an indication of a masquerade. The service also
7 validates the web site by appearance and behavior. The static content of the web site's
8 pages are checked; periodically the web site is challenged with a dummy transaction that
9 may or may not be known to the legitimate web site, and any behavioral abnormalities
10 noted as symptomatic of a masquerade.

11 Figure 1 shows the general disposition of the components of the subject invention. End
12 user workstation 1 contains browser 2 and proxy 3, such that browser web site access
13 requests are sent first to proxy 3 before being sent on to the Internet 5. Also shown in the
14 Figure are one of possibly many domain name service (DNS) servers 10, validation server
15 11, impostor website server 12 and legitimate website server 13. In normal use, the end
16 user uses user workstation 1 to access web sites on the Internet. The user invokes browser
17 2 and supplies the Universal Resource Locator (URL) for the web site.

18 Software in the user workstation (located e.g. in the proxy or, if arrangements have been
19 made with the browser vendor, in the browser) accesses the DNS server 10 to resolve the
20 URL's domain name component to an Internet Protocol (IP) address. In normal operation,
21 DNS server 10 returns the IP address of valid website 13 to the user workstation 1 so that
22 software in user workstation 1 (typically the browser) can then access valid website 13.

23 A site masquerade attack on valid website 13 typically consists of two components. First,

1 impostor website 12 accesses valid website 13 to obtain copies of valid content, so that
2 when impostor website 12 is accessed it will appear identical to valid website 13. Then,
3 impostor website 12 attacks DNS server 10 to update its tables in such a way as to cause
4 the domain name of the valid website 13 to be resolved invalidly to the IP address of the
5 impostor website 12. The details of the imposter's methods change from time to time and
6 are not part of the present invention, which is directed at thwarting imposters.

7 If the imposters's penetration is successful, attempted end user accesses to valid website
8 13 will resolve instead to impostor website 12; and, because impostor website 12 has
9 duplicated content from valid website 13, the end user will not be warned by any unusual
10 appearance or behavior.

11 With the subject invention, an access originating at browser 2 will access the DNS server
12 10 and resolve to the IP address of the impostor website 12 as before. However, in a
13 system according to the invention, when the website access request is sent from user
14 workstation 1 it will pass first to proxy 3, which will then contact validation server 11. As
15 a first inquiry, proxy 3 will ask validation server 11: 1) if the domain name has resolved
16 validly to the given IP address (of website 12) in the past, 2) if the IP address (website 12)
17 is known to be that of an impostor website, or 3) if there are significant differences in
18 behavior or appearance between data taken from valid website 13 and stored in validation
19 server 11 and corresponding data taken from impostor website 12 (whether stored in
20 server 11 or ascertained in response to this particular request). If the answer to alternative
21 1 is negative or the answer to either alternative 2 or 3 is positive, proxy 3 will notify the
22 end user of user workstation 1 of the probable impostor.

23 In the preferred embodiment, the user is free to access website 12 after the warning.

1 Figure 2 details the processing of proxy 3 of Figure 1. In block 20, the proxy waits for a
2 site request from the browser 2 of Figure 1. When that request is received, it is not
3 automatically sent on but is blocked, pending further processing by the proxy. In blocks
4 21 and 22, the domain name is captured, the DNS interrogated to obtain the resolved IP
5 address, and the address is captured as well. Block 23 packages these values into a
6 request to the Validation Server, shown as block 11 in Figure 1.

7 The Validation Server receives and processes this request and replies in a manner to be
8 described subsequently. In block 24 the response is received and analyzed to see if the
9 Validation Server has discovered problems with the domain name mapping or with the
10 site itself. If no problems are found, branch 26 is taken to block 32, which permits the site
11 request blocked previously to be sent on to the Internet.

12 If the Validation Server finds problems, branch 25 is taken to block 27, which takes the
13 Validation Server response and uses its contents to create a user dialog box. Block 28
14 then presents that dialog box to the end user and captures the end user's response. In the
15 case in point, the dialog box asks the end user whether to abort the site request or to allow
16 it to proceed.

17 This is not the only possible action, however. In other versions of the invention, the
18 dialog box may ask the user if the request should be altered to the valid site, or to report a
19 DNS error to the appropriate authorities, or may perform any other action consistent with
20 the determination of problems in the site access request by the Validation Server.

21 As will be subsequently described, the Validation Server itself may take some actions
22 when a site access request is detected to have problems.

1 If the user chooses to allow the site access to go forward, branch 31 is taken to block 32
2 whose function was formerly described. If the user chooses to abort the site access,
3 branch 30 is taken to block 20, which then awaits a subsequent site access request.

4 Figure 3 details the logic flow of the Validation Server. The Validation server is
5 preferably implemented as a Web Service. For details of Web Services, see the book Web
6 Services by Ethan Cerami, O'Reilly and Associates, published February 2002, ISBN
7 0596002246.

8 In Figure 3, processing starts with the receipt of an interrogation from an end user
9 workstation, generated in block 23 of Figure 2. The interrogation is received in block 40
10 of Figure 3. A series of tests are performed on the information in the interrogation, the
11 first of which is performed in block 41. In that block the pair consisting of a domain name
12 and an Internet address is tested against a list of such pairs kept locally in the Validation
13 Server. The maintenance of this list will be described subsequently. If the pair is not
14 valid, branch 42 is taken to block 50, which sends a negative response to the
15 interrogation. If it is valid, branch 43 is taken to block 44. Block 44 checks to see if the
16 Internet address is on a watch list, maintained locally in the Validation Server. The
17 maintenance of this list will be described subsequently. If the Internet address in the
18 interrogation is on the watch list, branch 46 is taken to block 50 whose function has been
19 previously described. If not, branch 45 is taken to block 47.
20 Block 47 tests the status of the requested site, as maintained by the Validation Server.
21 This test is conducted on the domain name of the site rather than on its Internet address,
22 as a safeguard against the case that the Validation Server does not have a correct mapping
23 of domain name and Internet address. This could occur if the domain name service used
24 by the Validation Server has been successfully attacked. The test is performed using a list
25 of domain names and their status maintained locally in the Validation Server. The

1 maintenance of this list will be described subsequently.

2 If the test fails, branch 49 is taken to block 50, whose function has been previously
3 described. If the test succeeds, branch 48 is taken to block 51, which sends a positive
4 answer to the interrogation from the end user workstation. After a response is sent,
5 whether positive or negative, block 40 is re-entered to await the next interrogation.

6 The list of pairs consisting of a domain name and an Internet address, maintained locally
7 in the Validation Server, is used to validate an interrogation request from an end user
8 workstation. Entries are created in this list typically when a site access request is received
9 from a user workstation, and the domain name in the access request is not present in the
10 list.

11 Typical processing in the Validation Server would be to contact one or more domain
12 name servers to obtain Internet addresses for the domain name, and to check for
13 agreement among the responses. If there is agreement or if there is substantial agreement
14 the site would then be contacted. If the site supports certificate-based authentication the
15 Validation Server would then authenticate the site and if it is found authentic a pair would
16 be created and entered into the list.

17 Authenticity can also be estimated from historical data. If there is already a pair in the list,
18 and if the Internet address obtained from the domain name service has changed, re-
19 authentication would be done.

20 The watch list consists of a list of domain names and Internet addresses, maintained
21 locally in the Validation Server. This list is used to check an interrogation request from an
22 end user workstation. Entries are created in this list typically when a message is received

1 from a recognized authority (e.g., government agency, Internet governance site) to the
2 effect that a masquerade may be in progress for a particular website. An entry may also be
3 made in this list when, in the process of checking an interrogation request, multiple
4 distinct responses are encountered when accessing the domain name system of the
5 Internet. The watch list may be shared with other instances of the Validation Server.

6 The status list consists of a list of domain names, each with an associated status. The
7 status may be either OK, indicating that the site is behaving normally, or not OK,
8 indicating that the site is behaving in a manner consistent with a masquerade. The status
9 may also be uncertain, indicating that although behavior has been substantially normal, it
10 is not now consistent with past behavior. The status may also be unknown, indicating that
11 the behavior of the site has not been determined, or has been determined so long ago that
12 it may no longer be valid. This list is maintained locally in the Validation Server.

13 The Validation Server determines site status with means including, but not limited to,
14 static content verification, behavior verification, capacity verification, verification means
15 agreed to with the site, or the exchange of certificates or other cryptographically encoded
16 information with the site.

17 Static content verification consists of typically retrieving web page content from a
18 number of web pages of that site, sampling the received content (e.g., images, text),
19 computing a hash code of that content and comparing the hash code with a previously-
20 stored hash code. Behavior verification consists of typically filling out a web form and
21 submitting it, possibly with intentional errors, and analyzing the resultant site behavior.
22 Behavior verification can extend to the actual purchase of an item from the web site,
23 while checking responses at each step. Preferably the item would be charged to a special
24 account maintained by the website, such that no shipments would be made and no charges
25 incurred. Capacity verification is a form of denial-of-service attack in which the

1 Validation Server submits transactions at a high rate and verifies the website's ability to
2 service these transactions. Capacity verification, while intrusive, depends on the
3 likelihood that legitimate websites have much higher capacity than masqueraders.

4 Verification means agreed to with the site include the above-mentioned purchasing
5 behavior, but may also include protocols known only to the legitimate website and to the
6 Validation Server. Finally, the exchange of certificates or other cryptographically encoded
7 information are well-known in the art and include IPSec and SSL.

8 The Validation Server may perform certain actions not shown in Figure 3, including the
9 automatic notification of appropriate authorities that a new masquerade may be occurring.
10 The Validation Server may be one of a number of like servers maintained by the same
11 service provider or by different service providers, such that a prior agreement exists
12 among these service providers to exchange information relating to masquerades.

13 As masquerades are illegal, the Validation Server may be obligated to report potential
14 masquerades to law enforcement agencies and possibly to Internet organizations
15 concerned with fraud, such as the CERT Coordination Center at Carnegie-Mellon
16 University.:

17 It can be seen that the description given above provides a simple, but complete
18 implementation of the automatic detection and foiling of a website masquerade. There
19 may be a concern as to an attack that masquerades as a validation server 11 of Figure 1.
20 Such a masquerade could cause an impostor validation server to fail to identify an
21 impostor website 12. Conventional techniques (e.g., Secure Sockets Layer -- SSL) can be
22 used to prevent such a masquerade, and only the real validation server 11 must implement
23 SSL.

1 Alternatively, a private validation protocol based on certificate exchange or any other
2 cryptographic or other technique could be used to protect against masquerades of the
3 validation server 11.

4 The description provided enables many forms of service provision. In one form a service
5 provider charges for each response from the Validation Server to a user workstation. In
6 another form the Validation Server charges legitimate websites for services provided to
7 end users for free.

8 Service providers may provide regionally specialized services or may specialize in certain
9 classes of websites, and charging may be contingent on the degree of certainty that a
10 particular website is legitimate.

11 In solving this problem, we rely on two characteristics of masqueraders: the fact that their
12 Internet addresses are the same as the addresses of legitimate sites, and the probability
13 that their behavior is not the same as that of legitimate sites.

14 Those skilled in the art will appreciate that the description above may be modified in
15 some details. For example, the functions of proxy 3 may be located on a server
16 maintained by an organization such as a corporation that has a firewall separating its
17 internal net from the Internet. The local IT department may choose to centralize the
18 functions of the proxy in the server that operates the firewall. Also, the functions of
19 validation server 11 could be performed by a server controlled by an organization that
20 employs the users, i.e. a private server as opposed to a server that accepts requests from
21 any user.

22 Also, the functions of the proxy could be performed by a browser or by an Internet

1 service provider.

2 If desired, the user might maintain a list of authentic Internet addresses, so that the

3 authentication process described above is not repeated for each access request. This list

4 constitutes a client-side cache of the contents of the Validation Server. Client-side

5 caching is known to those skilled in the art.

6 While the invention has been described in terms of a single preferred embodiment, those

7 skilled in the art will recognize that the invention can be practiced in various versions

8 within the spirit and scope of the following claims.